APPLICATION NOTE

AP-307

January 1987

# EPLDs, PLAs and TTL Comparing the "Hidden Costs" in Production

**PEDRO VARGAS**
PROGRAMMABLE LOGIC APPLICATIONS
INTEL CORPORATION

Order Number: 292030-001

## INTRODUCTION

When comparing logic alternatives, too often the outcome is dominated by the piece price of the components. A side by side comparison based on component costs only, may give the appearance that EPLDs are cost prohibitive. However, when the overall cost of manufacturing a system is considered, the higher integration of EPLDs proves to be a cost-effective solution.

## OBJECTIVE

This application note examines the total costs associated with designing, prototyping, and manufacturing a system. Once these costs have been examined, a comparison is made between EPLDs and other logic alternatives. By being aware of these additional costs, the engineer can make a more accurate cost comparison as a design is begun.

## COSTS DEFINED

Costs can be difficult to pinpoint, let alone measure. However, with a bit of examination, we can break down costs into the following categories;

- Design costs — the cost of conceiving a product
- Prototype costs — first implementation of the product idea
- Production costs — volume manufacturing of the product

Usually, the brunt of the cost for the first two categories is dismissed as NRE (non recurring expense). The effect of these costs on the overall project is examined later, let's look at the third category. Production costs, can be further broken down into;

- Component costs— the cost of the parts per board
- Inspection costs — labor costs for receiving the parts
- Inventory costs — the cost for storing, handling and dispensing the parts
- PCB fabrication — the cost for labor and equipment used in building a board
- Integration costs — the cost of harnesses, enclosures, nuts and bolts etc.

It's important to understand how the cost of a product is affected not only by the cost of the ICs used, but also by the other costs listed above. Figure 1 is a graph which shows this relationship.

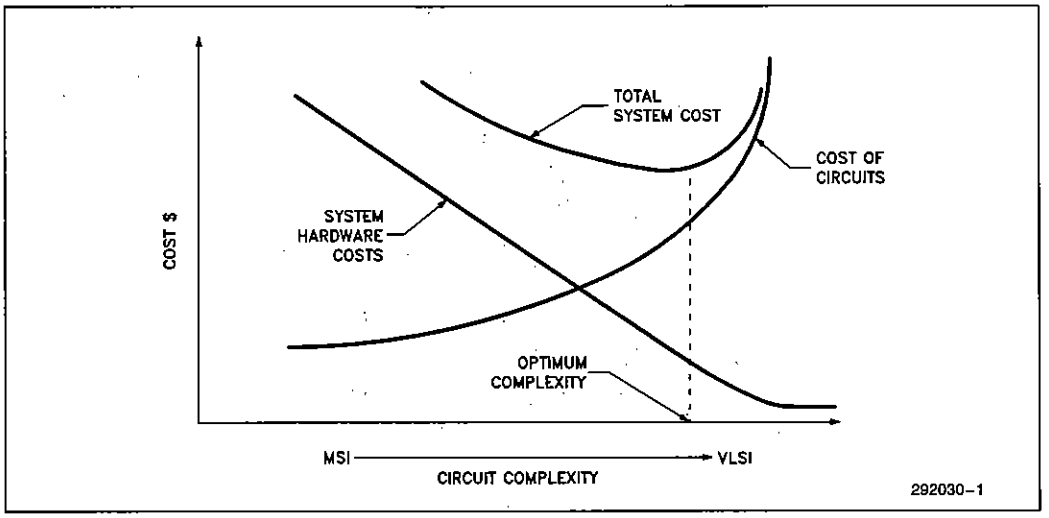


**Figure 1. Optimizing Circuit Complexity**

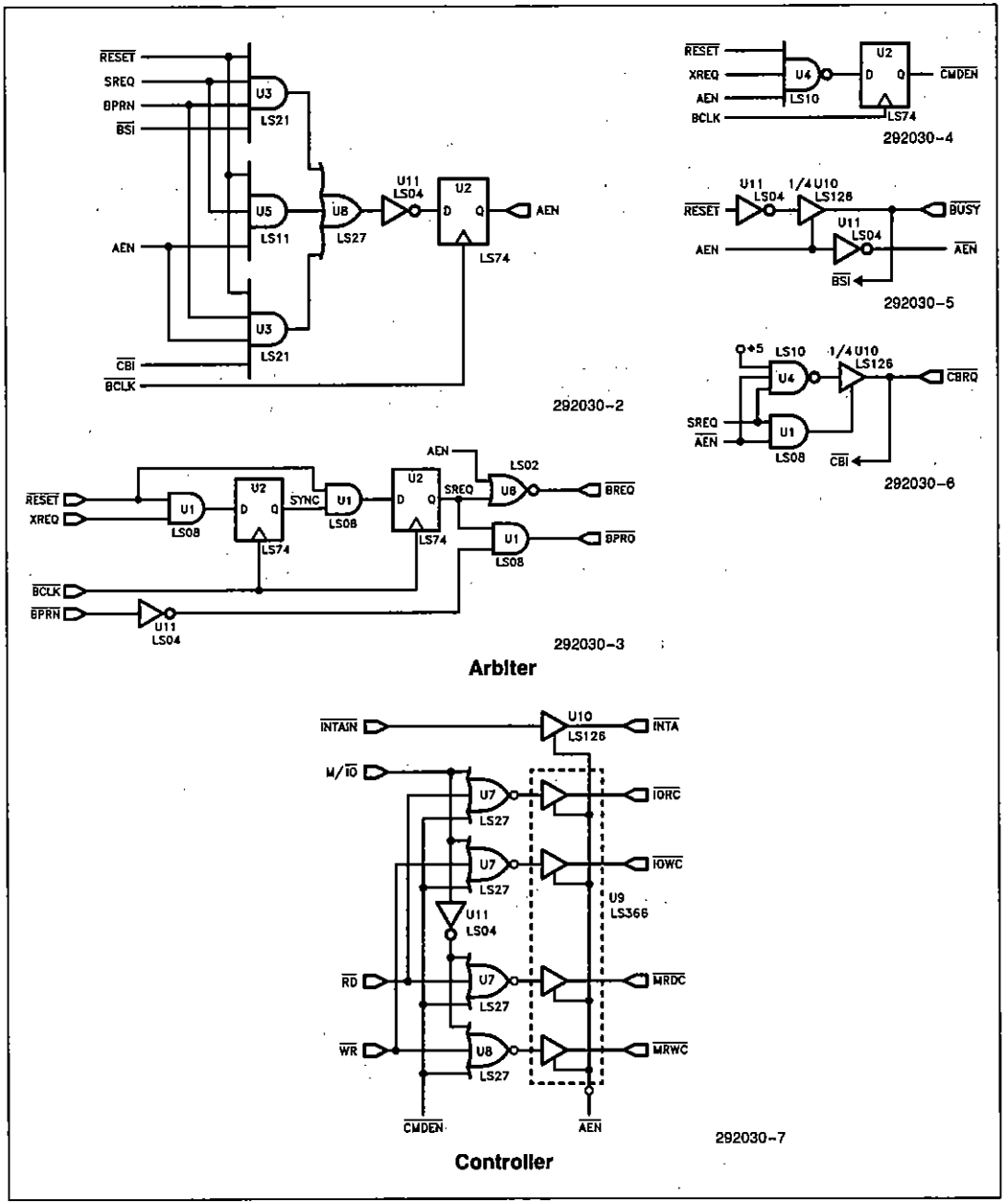


**Figure 2. MULTIBUS Arbiter/Controller–TTL Implementation**

The graph shows that as the density of the components used in a system progresses from SSI to VLSI, the cost for these devices increases. This isn't surprising, denser chips cost more to make. At the same time, by using denser devices, system hardware cost decreases. This is shown by the center line, which encompasses all the costs listed above. The bathtub curve above these shows the effect that denser ICs has on a system. That is, by using higher integration ICs, more functions are removed from the board. This in turn reduces the cost of the system in labor and parts costs.

A cost-effective product is one that uses the most efficient logic for the application. It's important to note that use of the least expensive component may not translate into system cost savings.

PAL* is a registered trademark of Monolithic Memories Inc.

## ARBITER CIRCUIT

Let's explore costs in more detail with an example. The example used here is the circuit of Figure 2, a MULTIBUS® I arbiter/controller. The circuit is used by bus masters arbitrating for control of the bus. Our implementation comparison contrasts TTL, PAL*, and EPLD solutions.

### Implementation Requirements

The TTL implementation is typical of many board level designs in the sense that it relies on inexpensive LSTTL. Figure 2 shows that the implementation is composed of standard logic gates and D-latches. The component list in Table 1 shows the circuit breakdown in more detail.[20]

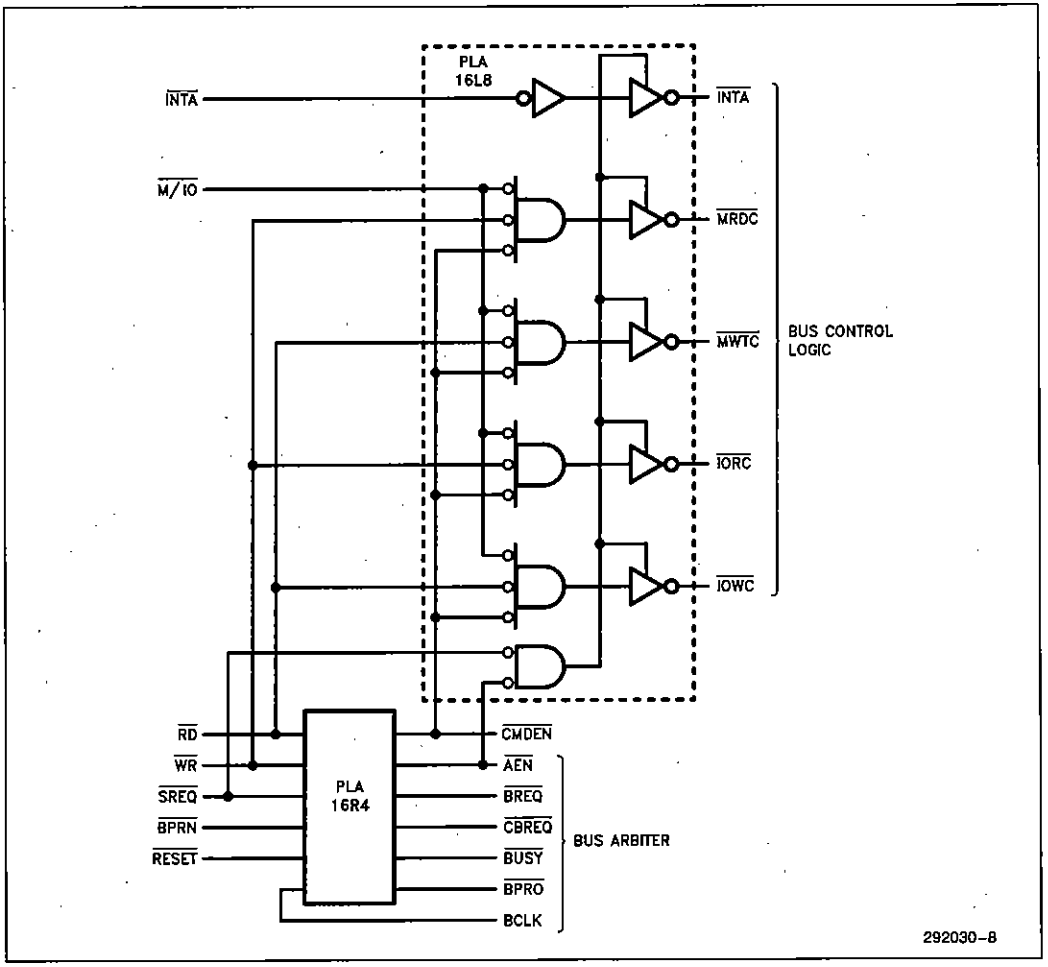


**Figure 3. MULTIBUS Arbiter/Controller-PAL Implementation**

**Table 1. Arbiter/Controller TTL Component List**

| IC | Type | DIP | $I_{CC}$ (mA) | Area ($In^2$) | Cost $ |
|---|---|---|---|---|---|
| U1 | LS08 | 14 PIN | 8.8 | 0.21 | 0.18 |
| U2 | LS74 | 14 PIN | 8 | 0.21 | 0.24 |
| U3 | LS21 | 14 PIN | 4.4 | 0.21 | 0.22 |
| U4 | LS10 | 14 PIN | 3.3 | 0.21 | 0.16 |
| U5 | LS11 | 14 PIN | 6.6 | 0.21 | 0.22 |
| U6 | LS02 | 14 PIN | 5.4 | 0.21 | 0.17 |
| U7 | LS27 | 14 PIN | 6.8 | 0.21 | 0.23 |
| U8 | LS27 | 14 PIN | 6.8 | 0.21 | 0.23 |
| U9 | LS366 | 16 PIN | 21 | 0.24 | 0.39 |
| U10 | LS126 | 14 PIN | 22 | 0.21 | 0.39 |
| U11 | LS04 | 14 PIN | 6.6 | 0.21 | 0.16 |

The PAL version of the circuit is shown in Figure 3. Two PALs are used due to the requirement of registered outputs on several of the signals.[20]

The complete circuit can also be designed in one 5C060 EPLD (Figure 4).[18] Looking at the three figures quickly points out the amount of circuit board space required by each version. The three implementations are compared side by side in Table 2.

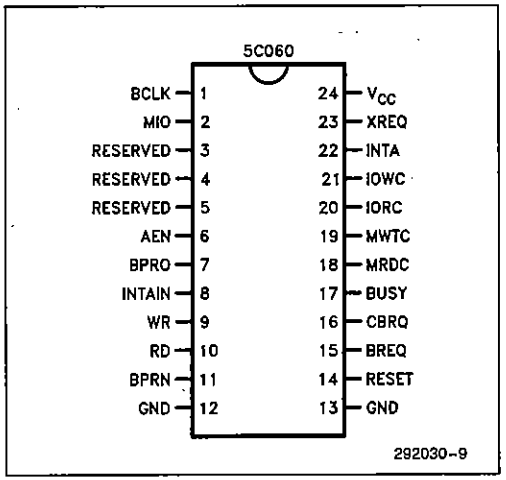


**Figure 4. MULTIBUS Arbiter/Controller-EPLD Implementation**

**Table 2. Implementation Results for Arbiter/Controller**

| Item | TTL | PLA | EPLD |
|---|---|---|---|
| IC Count | 11 | 2 | 1 |
| Pin Count | 156 | 40 | 24 |
| Interconn | 36 | 7 | 0 |
| Area | 2.34 | 0.6 | 0.36 |
| $I_{CC}$ (mA) | 100 | 240 | 15 |
| $P_{wr}$ (mW) | 500 | 1,200 | 75 |

- IC Count — The total chip count
- Pin Count — The total number of IC pins
- Interconnections — The traces required to connect logic gates together
- Area (inches-square) — The sum of the area of all ICs
- $I_{CC}$ (mA) — The current consumed while active
- $P_{wr}$ (mW) — Total power consumption at 5 VDC.

## Production Costs

Earlier, we noted that production costs consist of many variables. Usually, these variables are lumped together under the term "hidden cost". Although hidden costs are kept in mind by engineers, lack of tangible figures usually precludes their use in detailed cost breakdowns. For this reason, several manufacturers and consulting firms have come up with typical costs per IC and per pin.

For example, SOURCE III (San Jose, CA) reports in one of their studies that the manufacturing cost of a system translates to about 0.35 cents per IC pin. ICE Corporation (Scottsdale, AZ) and EDN magazine concur that the inserted cost of an IC is about $2 dollars. DATAQUEST also published a cost of about $2 to $4 per IC. While the data seems to be consistent, most engineers want to see for themselves how figures like these might be arrived at. The next sections provide insight into this process.

### COMPONENTS

The cost of the component is the easiest value to obtain. A quick call to a distributer or (at worst) a scan through the back of BYTE magazine (for TTL) gives us this cost. Table 3 shows the breakdown of component costs for each version of our MULTIBUS I circuit.

**Table 3. Average Component Costs**

| Package | TTL | PLA | EPLD |
|---|---|---|---|
| DIP14 | $0.25 | | |
| DIP16 | $0.35 | | |
| DIP20 | $0.55 | $1.50 | |
| DIP24 | | $2.90 | $6.00 |

The price of TTL has changed very little for the last few years[24] while EPLDs are dropping in price tremendously. PALs have also leveled off in pricing. Why? Figure 5 shows the life cycle curve of IC products used by the semiconductor industry. From the curve we see that TTL is in the stable range and prices are not likely to drop much more. PALs are also maturing and approaching a stable pricing range. EPLDs however, are in a growth area and historically this is

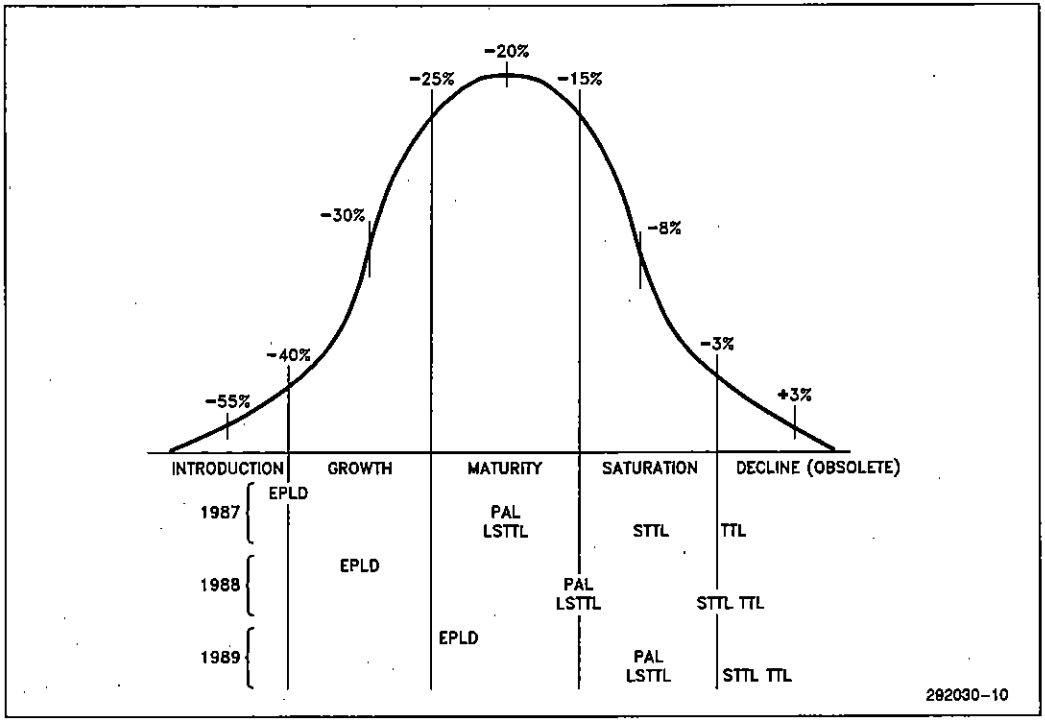


**Figure 5. Typical Price Changes Through Semiconductor Product Life Cycle**

where the heaviest pricing pressure is. This means that while EPLDs might be expensive (per part) right now, it's not out of the question to expect a 30% per year price reduction as the process is honed and perfected. In other words, it's also important to consider the price of a component at the projected production date, not just at design time.

Life cycle position is also important in understanding the gate cost that is associated with programmable logic devices like PALs and EPLDs. This relationship is shown in Figure 6. The curves translate our observation that newer devices have steeper price cuts during their introduction phase. The PAL curve shows that the cost per gate is leveling off due to the maturity of the device. In contrast, the EPLD is in the growth region, and based on the traditional price reductions, shows a cost per gate that intersects and bypasses the PAL curve.

## INCOMING INSPECTION

For most companies, incoming inspection is more than taking the parts and putting them on the shelf. Most have visual checking as well as some form of IC testing. The variables here are, what amount of human intervention is needed, are automatic handlers needed, are "go/no go" tests or "binning" done automatically? The typical scenario means that components are graded and tested individually, and then placed into one of several bins or kitted. Because the operators handle a large variety of pinned devices (resistors, capacitors, ICs), the cost can be distributed on a per pin basis. Many companies use a penny per pin for this cost.[16]

Inspection cost = $0.01 per pin

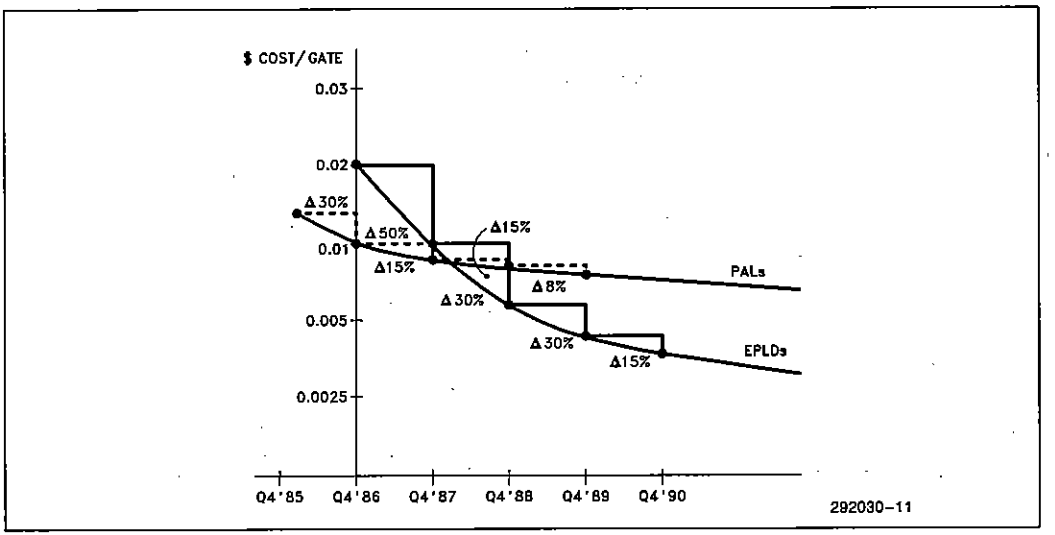


**Figure 6. Projected Cost Per Gate**

## INVENTORY

While most engineers agree that reducing parts count on their board makes the cost of inventory less, they usually attribute this to the reduction in component costs alone. In reality, the overhead of carrying inventory is made up of the following factors;[21]

- Cost of the component
- Cost of storage
- Maintenance costs
- Data processing
- Usage
- Taxes insurance and interest
- Turnover rate

The American Production and Inventory Control Society (APICS) reports that since 1973 the median cost of carrying inventory has been about 25% of total production costs. They also note that the largest contributing factors are the cost of materials handling storage, and data processing. For simplicity, let's limit our inventory cost to these items.

Inventory cost = storage + maintenance + processing

Depending on the locale of a company, the cost of storage can vary greatly. However, this cost is charged on a square foot per year basis. Lets assume a conservative figure of $20 dollars and distribute this amoung the ICs in our example circuit.

storage = [Total IC area (sq. ft.) × $20]/IC count

Maintenance refers to the cost of handling, counting, marking, and auditing each IC. Each production manager has their own way of keeping tabs on this. One way is to charge on a per part basis. A review from several production oriented journals cites $0.3 cents as the typical handling charge for 16 pin devices.[23]

Maintenance = $0.03 per 16 pin part.

Processing[21] usually entails a parts log that tracks each part by manufacturer, cost, second source etc. Also, monthly shortage reports are quite common as are quarterly orders and audits. Limiting this cost to paper only, at one sheet of paper per week, per year, at a cost of a penny per part type;

Processing = $0.52 per part type per year

## PCB FABRICATION

The cost of manufacturing (cutting, etching, drilling) a circuit board seems to vary around two pricing methods. Some fab houses charge on a square inch basis. Others base their price on a gut feeling based on previous jobs. The square inch method is the most common.

Items of interest in evaluating PCB costs are, number of ICs, number of traces and vias, and in general, the complexity of the board. Traces that are smaller than 10 mils require extra care in etching. Depending on complexity, and additional charge might be added to the area cost. This charge covers material loss in case of low etch yields. Yield is directly dependent on the number of ICs on a board. In other words, more ICs mean more holes, tighter traces, and a greater chance of losing some boards in their processing. The average going

rate is $0.20 cents per inch for double-sided boards. The price increases by about 40% for every two layers. This extra charge, however is too subjective to consider in our comparison.

PCB Fab = [$0.20 × total IC area (sq. inch)]/IC count

### Traces

There is a real cost involved with traces, which doesn't surface until later in the production cycle or on a later board revision. A technical paper presented at the 1984 international Test Conference[1] estimates that the cost of a trace on a board is ten to thirty times that of one made in silicon. The cost of traces is taken up by:

- Increased drilling (more traces = more vias = more holes)
- Lower PCB yield (smaller mill lines drop the board yield)
- Increased risk of trace to trace shorts (lower reliability)
- More expensive artwork mods (it costs more to move traces around on a board)
- More expensive PCB mods (cost of cuts, jumpers, and rework)

In our circuit example, an extra trace is that which is unnecessary in contrasting implementations. For example, referring to Figure 2, of all the traces required to connect/RESET in the TTL implementation, only one will be required for the EPLD and PAL circuit (the input); the others won't be needed.

For our comparison, let's take the median value of twenty as our multiplying factor. Since a silicon trace costs an order of magnitude less than an EPLD gate ($0.01), the resulting cost of a PCB trace is;

($0.01/10) × 20 = $0.02 cents per trace

Trace cost = [total trace count × $0.02]/IC count

### ASSEMBLY

The cost of assembling a board is largely dependent on labor charges and capital. Assembly consists of lead forming, component insertion, and soldering. The labor charge is hourly and varies between domestic and offshore assembly houses. While machines can certainly do lead cutting, crimping, and insertion, human intervention is still an expensive presence. Assembly costs can be charged on a per board or per chip basis. The latter is more appropriate for our comparison. The average charge (domestically) is about $0.10 per IC.

Assembly = $0.10 per 16 pin part

One important result of using high integration parts like EPLDs is that the assembly procedures (manual or automatic) go smoother. This is due to fewer parts being handled, and less overheating of the equipment. Overall, the industry reports less insertion faults (parts stuffed wrong) as denser ICs are used and as insertion equipment matures with them.

### TEST

Test strategies can vary, but the typical test flow for a board[3] is shown in Figure 7. The process is basically taking a board through increasing complexity levels of testing. For example, ATE might be a bed of nails fixture that catches 60 percent of the faults. Test bed is usually a backplane with all boards known good except for the one under test. System test is the final integration of all the boards that were tested individually.

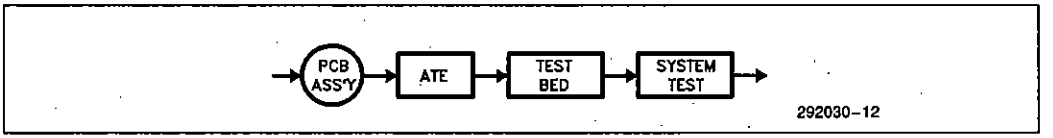


**Figure 7. Typical Test Flow**

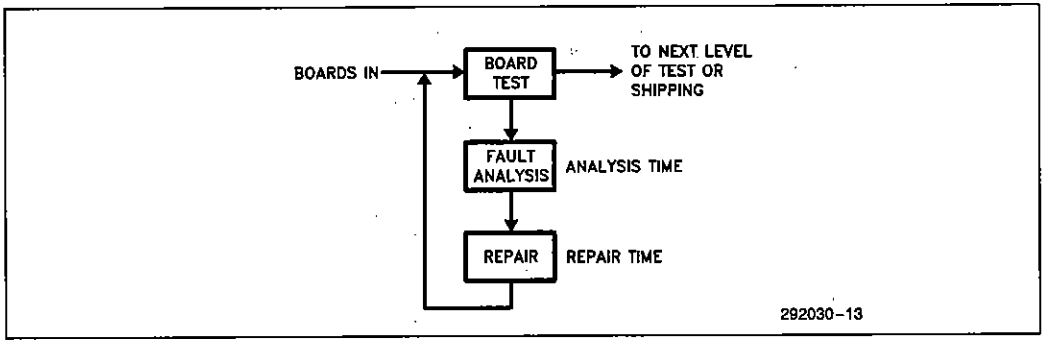


**Figure 8. Typical Test and Repair Loop**

Errors can occur at any step of the test flow; each time this happens, a test loop is initiated. This loop is depicted in Figure 8. The cost for testing a device depends on the cost of the equipment, depreciation, the labor rate, and other factors that are company dependent. There are several ways to reduce test costs, but the best way is to reduce the probability of errors occuring. There is no question that as the number of ICs increases, so does the probability of error.

With all things considered, the industry reports a nominal test cost of about $0.15 per IC.[27][28]

Test cost = $0.15 per 16 pin IC

## REWORK

The cost of rework is best understood by considering the cause of errors in more detail. Errors are typically caused by poor board quality, inadequate solder process, tolerance of insertion, and of course, bad chips. Table 4 shows the average board fault spectrum. The figures are a conclusion reached by EVALUATION ENGINEERING magazine[10] as to what the industry is currently seeing. The table shows that the majority of board errors is due to solder shorts. These errors are the result of traces or IC holes being too close, which is what happens on densely populated boards.

**Table 4. Average Board Fault Spectrum**

| | |
|---|---|
| Tolerance | 20% |
| Shorts | 40% |
| Insertion | 30% |
| Bad Parts | 10% |

Of all the material costs associated with rework, the main cost is the time spent on a repair. Considering that it takes approximately two minutes to desolder, insert, resolder, and clean a component pin[9], one can see that more ICs on a board directly affect cost. Repair times also increase dramatically on multi-layer boards that might have been doubled sided if denser logic was used.

For our comparison, let's assume that our test equipment is 95% efficient in finding solder faults on the first pass (no loop). This leaves 5% of the faults that go undetected and eventually must be found and repaired. The estimated cost per pin based on a $6.00 hourly wage and the two minute repair time is approximately $0.02 cents.

Rework = [$0.02 × total pin count]/IC count

It is important to note that the probability of errors is based on a Poisson distribution[8] that increases exponentially with the number of pins and components. This distribution is used in wave solder processing to correct for solder errors. Mathematically this is expressed as:

$$P = \frac{e^{-np}(np)^x}{XI}$$

where; P = The probability that a defect will occur
n = The number of components
p = The fraction defective
x = The actual number of defects

This means that the TTL and PAL version of the arbiter have a higher probability of error than the EPLD version. However, to make our comparison easier, let's simplify this to more of a linear relation. For each implementation, the rework cost per IC is calculated by;

Rework cost = [(total pin count) × (5%) × ($0.02 cents)]/IC count

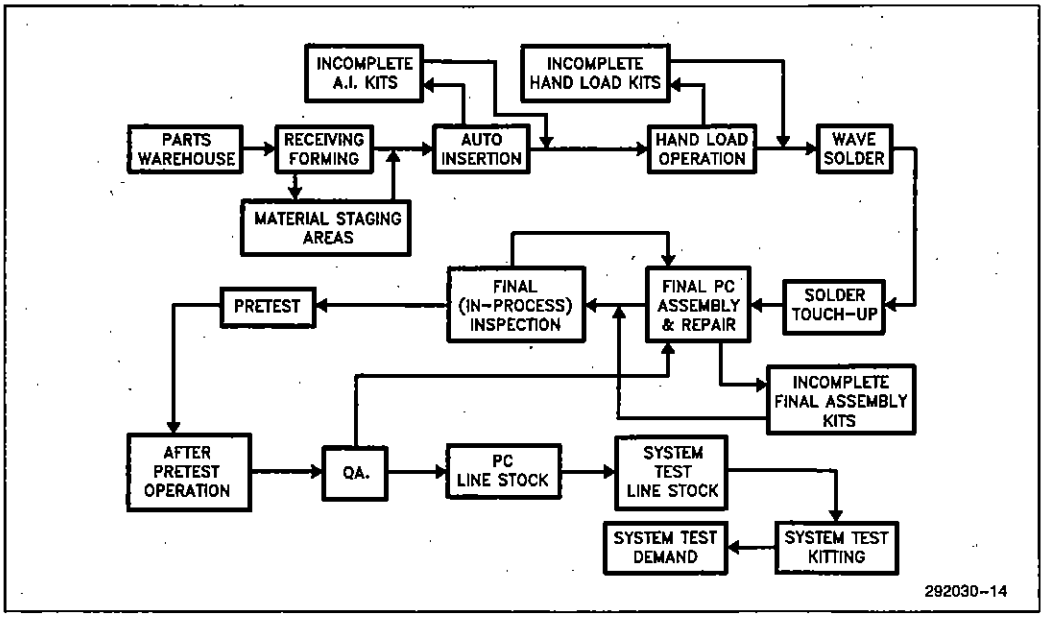


**Figure 9. Example of a Production Line**

### QUALITY CONTROL

In most production operations, boards go through several steps of quality inspection. The bare board might be inspected after preliminary tests and after system tests. Although 100% inspection should theoretically eliminate all errors, in real life this rarely happens. The main reason for this is the complexity of the production and rework loops as shown in Figure 9.

Quality control's purpose is to remove defective products and either junk them or rework them, neither of which is cost effective. The best approach is to design the quality in, not fix it in. One way to design in quality is by reducing the possibility of errors and increasing the reliability of a product. This is one of the primary advantages of dense logic (like EPLDs and PALs) over TTL.

A survey conducted by *CIRCUITS MANUFACTURING* magazine[8] yielded the cost of $10 to $50 dollars to inspect, find, and repair a defect on a board. They summarized that the actual cost of inspection is about $0.004 for each hole on a board. With this in mind, let us assume a 100% inspection of our arbiter circuit for each implementation. This means that each pin (and every trace via) will have to be looked at. The calculation for this is;

$$\text{QC cost} = (\text{total pin count} \times \$0.004)/\text{IC count}$$

### POWER SUPPLY

Price for 5V, single output, switching power supplies as advertised by several vendors is $1.00 per watt. The calculation for determining power supply costs in our comparison is:

$$\text{Power cost} = [(5\text{VDC} \times I_{CC}\ (\text{mA})) \times \$1.00\ \text{per watt}]/\text{IC count}$$

## Additional Costs

In addition to the more obvious costs, there are several other items that contribute to the "hidden cost" of a system.

### PROGRAMMING LOSS

Because PALs are a one time programmable type of device, full testing can't be done on them without destroying the user's fuses. For this reason PALs have a published programming loss of 2%[20]. The cost for this is:

$$\text{Programming loss} = (\text{PAL IC count} \times 0.02) \times \text{PAL cost per IC}$$

EPLDs, because they are based on EPROM cells, can be programmed for different patterns, fully tested before customer delivery, and then erased. The result is a near 100% percent programming yield[22].

## PROGRAMMING FEE

Programming fee is the cost of programming a device. While many companies have in-house programmers, it is quite common for programming to be done by the distributor. In some cases, and at low volumes, the programming may be done free of charge. However, at larger volumes a programming charge is not uncommon. The charge varies with volume, programmer availability and in general, your state of affairs with the distributor. The cost for programming EPLDs and PALs is the same per device and averages about $0.25 cents.

Programming fee = $0.25 cents

## SAFETY STOCK

Although this particular item was not mentioned in the inventory section, it plays a very important role in the production world. Safety stock[21] is extra ICs ordered to cover for unexpected events. Unexpected here might be a large unforeseen customer order or simply a bad batch of parts.

While industry seems to strive for the optimum JIT (just in time) production[14][16], which stresses minimal inventory until needed, it's not unusual for production managers to carry a five to ten percent inventory buffer depending on the cost of the part. In most cases, the larger expensive parts like microprocessors, peripheral controllers, and other LSI devices are safety stocked in smaller quantities.

Let's assume that the safety stock is to be a maximum of 10%. Five percent might be used to cover for the unexpected occurrences, and five for WIP (work in process) modifications. Since all parts have the same probability of unexpected events we can assign that percentage equally. Justifying the second 5% depends on the IC technology itself. For instance, WIP modifications usually require cuts and jumpers on TTL, therefore it's unnecessary to order the additional 5%. In process modifications to an EPLD are done simply by reprogramming it, here again there is no need for the additional 5%. PALs however cannot be cut and jumpered (internally) nor can they be reprogrammed. Also, there is the possbility that "on the shelf" PALs will be programmed in advance, therefore a WIP mod that impacts their function means that those parts must be obsoleted (junked). In this case, an additional 5% is justifiable.

Let us assume that the production manager reduces safety stock by a moderate amount, let's say 3%. In a case like this, usually the larger more expensive parts are curtailed first. Since EPLDs provide good coverage for work in progress and because they are more expensive by comparison, we can reduce the total safety stock to 2% and not compromise our safety margin. Because TTL is inexpensive it tends to suffer more of the "gunshot" approach in testing[7]. This means that the useage rate is greater because production technicians tend to replace TTL parts with more liberty. For this reason let's leave the TTL safety stock as it stands. PALs could be reduced, but faced with the fact that the programming yield is 2% and that internal modifications can't be made, the production manager might decide not to change the safety stock for PALs. These results are shown in Table 5.

**Table 5. Safety Stock**

| | TTL | PAL | EPLD |
|---|---|---|---|
| Unexpected Events | 5% | 5% | 2% |
| WIP MODS | 0 | 5% | 0 |
| Total | 5% | 10% | 2% |

The safety stock calculation for each implementation is:

Safety stock = (% of stock × IC type × IC type cost)/IC count

## DE-COUPLING CAPACITORS

While adding caps solves many problems due to system noise, it also increases the cost of PCB layout, PCB fab, and adds an additional burden on all of our other costs. For a TTL system, a good de-coupling rule of thumb is to use one 0.01 μf per each synchronous driven gate and at least 0.1 μf per 20 gates regardless of synchronicity. Engineers recognize the need for decoupling and usually take it a step further by using one capacitor per IC. Most boards reflect this practice, which, in itself is very good. However, the addition of all these caps is definitely measurable, in both component and systems cost.

The average cost of a ceramic capacitor in moderate quantities is about half a cent. For our comparison we will follow the accepted practice and de-couple each TTL, PAL, and EPLD device. Our capacitor cost is then:

De-coupling cost = $0.005 × IC count

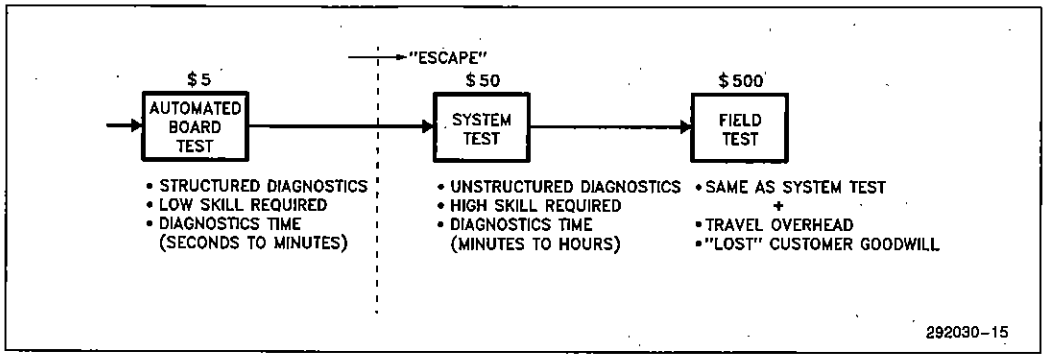


**Figure 10. Escape Costs**

## Other Costs To Consider

Eventually, some place toward the end of a production line, a board becomes part of a system. At this point it is housed in an enclosure and all the necessary cabling is done. Even here, however, the impact of using a particular IC technology can still be felt.

### DEFECT ESCAPES

One very significant item that the test community acknowledges is the cost of "escapes"[4]. "Escape" is defined as a fault that goes through the early stages of board test undetected. Figure 10 shows the escape relationship. An industry rule of thumb states that the cost to detect a fault increases by an order of magnitude at each stage. This means that if it costs $5 to find a fault at the board test level, that same fault might cost $50 at the system level and $500 at the field level. An important relationship to remember, is that the number of faults per board increases logarithmically, as the number of components on the board increases[6]. The cost of an "escape" is difficult to quantify, but generally, a board with a higher component count has a greater cost[2][8].

### CABLES/WIRING HARNESS

When the number of components or the power requirements of a system are reduced, a reduction in cables and wiring is usually expected. The cost savings here is either in the elimination of cables (because more functions are condensed into an IC) or the reduction of cable gauge or length (because less power is required, in the case of EPLDs). Also, fewer cables means fewer cable ties, connector pins, and mounting hardware. While this is a subjective figure, lets assume that the distributed cost of system cables is $0.25 per IC.

$$\text{Cable cost} = \$0.25 \times \text{IC count}$$

### ENCLOSURE

Certain applications require reduced packaging or enclosure size. In industrial control for example, each line might require a complete system to monitor it's operation. In a case like this, a large bulky box full of boards might not be appropriate. A good example of the benefits that high integration logic provide enclosures, is the third market versions of the popular PC. Many of these companies have fully compatible versions that fit on a single board. EPLDs and PALs are capable of providing a cost savings in this respect. However, while PALs approach the density requirements, their large power needs render them counterproductive to the low power specs of small systems. TTL is just not as effective as either PALs or EPLDs.

For our comparison let us assume the cost of enclosure per chip is $0.75. The calculation is:

$$\text{Enclosure cost} = \$0.75 \times \text{IC count}$$

Table 6 shows the cable and enclosure costs for the MULTIBUS I circuit. Although the results are based on assumed values, we can see that a larger IC count influences the burdened cost of the system. Our final comparison will not use these figures, but they should be considered.

**Table 6. Other Production Costs for Multibus I Circuit**

| | TTL | PLA | EPLD |
|---|---|---|---|
| Wiring/harness | $2.750 | $0.500 | $0.250 |
| Enclosure | $8.250 | $1.500 | $0.750 |

## Arbiter Circuit Conclusion

A compilation of the cost variables for our comparison is shown in Table 7a and 7b. Because the cost may differ for each company, the comparison calculations were done on a Lotus 1-2-3 worksheet that the individual engineer can modify with their specific values. The worksheet is available, and can be downloaded from the Intel EPLD bulletin board. Table 8 shows our calculation results for three years of production.

**Inventory:**

| | Costs |
|---|---|
| Incoming insp. ($/pin) | $0.010 |
| Storage ($/sq.ft./yr) | $20.000 |
| Maintenance ($/part) | $0.030 |
| Processing ($/part type/yr) | $0.520 |
| Safety stock (%) | 2% |

**Manufacturing:**

| | Costs |
|---|---|
| PCB fab. ($/sq.in.) | $0.200 |
| Assembly ($/part) | $0.100 |
| Test ($/part) | $0.150 |
| Rework ($/pin) | $0.020 |
| QC ($/pin) | $0.004 |
| Power ($/watt) | $1.000 |
| Interconn | $0.020 |
| Program ($/part) | $0.250 |
| Caps. (each) | $0.005 |

(a)

**Integrated Circuits**

**Component Count:**

| Package | TTL | PLA | EPLD |
|---|---|---|---|
| DIP14 | 10 | | |
| DIP16 | 1 | | |
| DIP20 | 0 | 2 | |
| DIP24 | | | 1 |

| ICs | Types |
|---|---|
| TTL | 10 |
| PLA | 2 |
| EPLD | 1 |

| Circuit Requirements: | $I_{CC}$ (max) | Interconnects |
|---|---|---|
| TTL circuit (total mA). | 100 | 36 |
| PLA circuit (total mA). | 240 | 7 |
| EPLD circuit (total mA). | 15 | 0 |

(b)

**Tables 7a and b. Multibus Arbiter/Controller Cost Variables**

**Table 8. MULTIBUS I Arbiter/Controller Production Costs**

| AVERAGE COMPONENT COST | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Year 1 | | | Year 2 | | | Year 3 | | |
| Package | TTL | PLA | EPLD | TTL | PLA | EPLD | TTL | PLA | EPLD |
| DIP14 | $0.25 | | | $0.20 | | | $0.19 | | |
| DIP16 | $0.35 | | | $0.30 | | | $0.27 | | |
| DIP20 | $0.55 | $2.00 | | $0.38 | $1.70 | | $0.35 | $1.56 | |
| DIP24 | | | $6.00 | | | $4.20 | | | $2.90 |
| **PRODUCTION COSTS** | | | | | | | | | |
| | Year 1 | | | Year 2 | | | Year 3 | | |
| Item (costs per part) | TTL | PLA | EPLD | TTL | PLA | EPLD | TTL | PLA | EPLD |
| Components | $0.259 | $2.000 | $6.000 | $0.209 | $1.700 | $4.200 | $0.197 | $1.560 | $2.900 |
| Incoming Insp. | $0.142 | $0.200 | $0.240 | $0.142 | $0.200 | $0.240 | $0.142 | $0.200 | $0.240 |
| Inventory | | | | | | | | | |
| Maintenance | $0.027 | $0.038 | $0.045 | $0.027 | $0.038 | $0.045 | $0.027 | $0.038 | $0.045 |
| Storage | $0.030 | $0.042 | $0.050 | $0.030 | $0.042 | $0.050 | $0.030 | $0.042 | $0.050 |
| Processing | $0.473 | $0.520 | $0.520 | $0.473 | $0.520 | $0.520 | $0.473 | $0.520 | $0.520 |
| Printed Circuit Board | | | | | | | | | |
| Fabrication | $0.043 | $0.060 | $0.072 | $0.043 | $0.060 | $0.072 | $0.043 | $0.060 | $0.072 |
| Trace costs | $0.065 | $0.070 | $0.000 | $0.065 | $0.070 | $0.000 | $0.065 | $0.070 | $0.000 |
| Assembly | $0.089 | $0.125 | $0.150 | $0.089 | $0.125 | $0.150 | $0.089 | $0.125 | $0.150 |
| Board test | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 |
| Rework | $0.014 | $0.020 | $0.024 | $0.014 | $0.020 | $0.024 | $0.014 | $0.020 | $0.024 |
| QC | $0.057 | $0.080 | $0.096 | $0.057 | $0.080 | $0.096 | $0.057 | $0.080 | $0.096 |
| Power Supply | $0.045 | $0.600 | $0.075 | $0.045 | $0.600 | $0.075 | $0.045 | $0.600 | $0.075 |
| Total Cost/Part | $1.393 | $3.904 | $7.422 | $1.343 | $3.604 | $5.622 | $1.331 | $3.464 | $4.322 |
| | | | | | | | | | |
| Total Cost/System | $15.321 | $7.808 | $7.422 | $14.771 | $7.208 | $5.622 | $14.641 | $6.928 | $4.322 |
| Additional Costs/System | | | | | | | | | |
| Programming loss | $0.000 | $0.080 | $0.000 | $0.000 | $0.068 | $0.000 | $0.000 | $0.062 | $0.000 |
| Safety stock | $0.143 | $0.400 | $0.120 | $0.115 | $0.340 | $0.084 | $0.109 | $0.312 | $0.058 |
| Programming fee | $0.000 | $0.500 | $0.250 | $0.000 | $0.500 | $0.250 | $0.000 | $0.500 | $0.250 |
| De-coupling caps | $0.055 | $0.010 | $0.005 | $0.055 | $0.010 | $0.005 | $0.055 | $0.010 | $0.005 |
| True mfg. cost/system | $15.518 | $8.798 | $7.797 | $14.941 | $8.126 | $5.961 | $14.804 | $7.813 | $4.635 |

The comparison in component costs shows that the EPLD costs more than either a TTL or PAL IC. As costs are added, the figures for TTL and PALs begin to approach the cost of an EPLD. These are shown on the line labeled "Total cost/part".

The "Total cost/system" line shows the actual cost when all the ICs are considered. For the first year, the TTL version is the more expensive implementation, and the EPLD numbers look very favorable.

The "True mfg. cost/system" line results after additional costs are figured in. Here we see that the first year, the EPLD version already provides a $1 savings over the PAL version, and that the cost of the TTL implementation is very high. Also, the inserted cost per IC at this point is, $1.15 for TTL, $2.40 for PAL and $1.80 for the EPLD. This is in line with the inserted costs that we mentioned earlier.

The production costs for two additional years shows that the decreasing price of EPLDs (based on the curve of Figure 5) will continue to provide costs savings as production ramps up in quantities.

In terms of functional benefits, the EPLD implementation is the most beneficial because;

- The chip count has gone down, one EPLD has replaced 11 TTL ICs in one implementation, and 2 PALS in the other, reducing the cost and time of:
  —board layout
  —board fab
  —assembly
  —rework
- The reliability of the board has increased. Fewer components translates into less probability of error.
- Modifications are easier to make. Instead of cuts and jumpers (for TTL), or throwing away a PAL, a change is re-programmed.
- The need for de-coupling caps is reduced. All those individual ICs are eliminated and in some cases the distributed capacitance of the board may be enough de-coupling.
- Power supply requirements are small. The active current requirements are much smaller with EPLDs. This in turn reduces the need for large power supplies and fans.
- Cable requirements and enclosure benefits have been improved. Since EPLDs provide better integration over TTL and PALs, the size of the system will be smaller. This translates into fewer boards and cables.
- Inventory is reduced. One EPLD replaces many TTL devices. Also, "on the shelf" programmed EPLDs can be reused in a pinch, PALs can't.

Less expense and probability of "escapes". The time and cost of finding and fixing escape problems is reduced to one reprogrammable IC. In the field, this translates into less "down time" for the customer and a higher level of customer "goodwill" for the OEM.

Allows capability for customized hardware. Specific customer requirements can be implemented. Also, DIP switches and configuration jumpers may not be necessary in many cases, since configurations can be programmed into the EPLD.

## Development Costs

As mentioned earlier, the costs of development are usually dismissed as NRE. One reason for this is the difficulty in pegging down these costs. However, while money might be expendable at this stage, time is usually critical. Time saved at the front end can make a difference in beating the competition to market. The following topics are presented for consideration. No costs are assigned to them.

### RESEARCH

The amount of time spent researching components, component sources, and technical data can be very large. Designs done with a large IC count require more research and analysis time. Higher integration devices require learning curve time, but, in the long run this tends to reduce research time, especially in future designs.

### PROTOTYPING

For most companies, prototypes are three to five level wire wrap boards built by inhouse technicians or outside contractors. During prototype fab, a certain amount of work has to be done to each IC. Part of this work is, adding bypass caps, labeling chips, and lead forming. In smaller companies, the board might be hand wrapped. Larger companies might use an automatic wrapper. Once the board is wrapped, a continuity check is done on each wire net to insure connections and minimize shorts.

The turn around time for a protoboard is one to two weeks and can be shortened by paying a premium price. An alternate way of shortening this time is to simplify the board by using denser ICs.

### DEBUGGING

Fixing bugs on a protoboard involves unwrapping and wrapping connections, as well as replacing ICs. Making mods on a TTL board is very time consuming and error prone due to the large numbers of wires. Making mods with PALs is expensive since the part usually has to be junked. EPLDs in contrast, are re-programmable and lend themselves to all the revisions that are common in the early design stages.

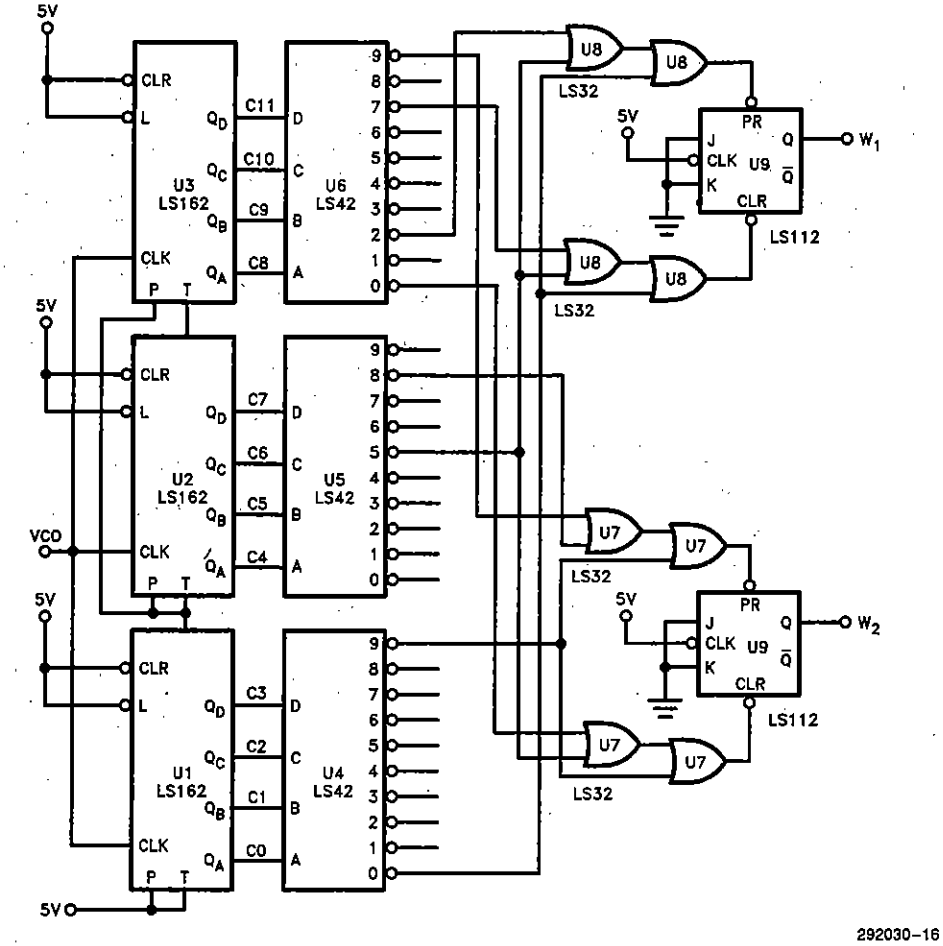


**Figure 11. Time Window Generator, TTL Circuit**

### PCB LAYOUT

Artwork quotes are based on several factors. These are, board size, number of 16-pin chip equivalents, pad count, and the chip to board packing ratio. The chip equivalents are calculated by taking the total lead count (ICs and discretes) and dividing by 16. Pad count is the number of holes in the board. The packing ratio determines how much room an IC has around it. This is critical because space is needed to place sockets, vias, and trace bends. Currently, most service bureaus consider 0.75 square inches per IC to be the minimum packing density. This figure applies to DIPs only, other packages like SMT (Surface Mount Technology) will improve on this. However, for standard DIPs anything less than this might push the board into a multi-layer.

During schematic evaluation, the bureau doesn't usually charge for traces directly. Because they can't foresee the exact count, and they don't have time to count them on the sheets, they make a judgment based on previous jobs. If the board appears to be tight, their autorouter (CAD based) won't be as efficient, and more hand layout will have to be done. However, as more CAD based service bureaus integrate schematic capture front ends, the cost of traces and vias will be more visible.

Because the evaluation is subjective, the final cost varies, and is a combination of charges. However, because pad count can be determined easily, the overall price is usually gauged against a pad price.

## WINDOW CIRCUIT

### Background Information

In applications that involve time-division multiplexing, it is useful to have a circuit that windows a specific area of the bit stream [27]. The circuit of Figure 11 is a TTL implementation of such a circuit. The idea is to count time slots from a known reference and at a certain decode, set and clear a latch. The output of the latch is the time window, which might be used for further gating in other parts of the circuit. The TTL parts list is detailed in Table 9.

The PAL alternative of Figure 12 is comprised of two 16L8s and one 16R4. While the component count has been reduced from nine to three, there are still fourteen extra interconnections.

One 5C060 is needed to integrate the complete circuit. Fourteen out of the sixteen EPLD macrocells are used, and exernal traces are only the three I/O pins as shown in Figure 13.

### Production Costs

The production variables for the window circuit are shown in Table 10a and 10b, and the production costs in Table 11. The comparison shows three years of system costs for each implementation.

**Table 9. TTL Component List for Window Generator**

| IC | Type | DIP | $I_{CC}$(mA) | Area(in$^2$) | $ |
|---|---|---|---|---|---|
| U1 | LS162 | 16 | 32 | .24 | .49 |
| U2 | LS162 | 16 | 32 | .24 | .49 |
| U3 | LS162 | 16 | 32 | .24 | .49 |
| U4 | LS42 | 16 | 13 | .24 | .39 |
| U5 | LS42 | 16 | 13 | .24 | .39 |
| U6 | LS42 | 14 | 13 | .24 | .39 |
| U7 | LS32 | 14 | 9.8 | .21 | .18 |
| U8 | LS32 | 14 | 9.8 | .21 | .18 |
| U9 | LS112 | 14 | 6 | .21 | .29 |

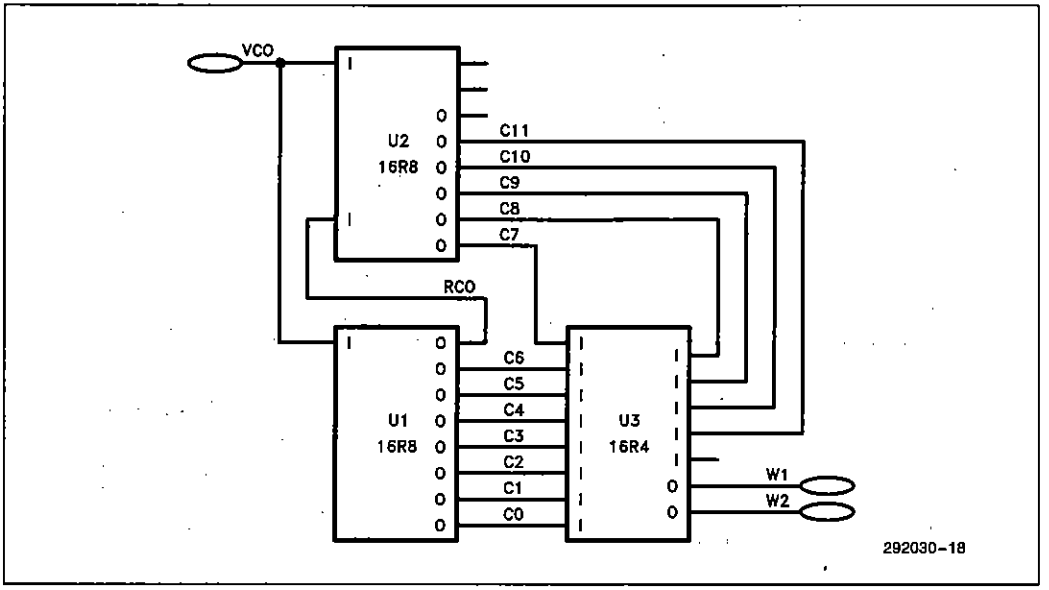


**Figure 12. Time Window Generator, PAL Circuit**

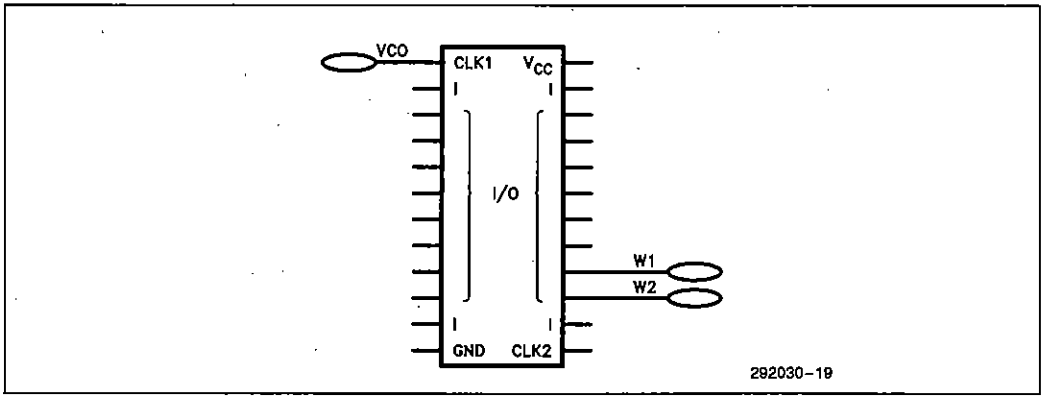


**Figure 13. Time Window Generator, EPLD Circuit**

| Inventory: | Costs |
|---|---|
| Incoming insp. ($/pin) | $0.010 |
| Storage ($/sq.ft./yr) | $20.000 |
| Maintenance ($/part) | $0.030 |
| Processing ($/part type/yr) | $0.520 |
| Safety stock (%) | 2% |

| Manufacturing: | Costs |
|---|---|
| PCB fab. ($/sq.in.) | $0.200 |
| Assembly ($/part) | $0.100 |
| Test ($/part) | $0.150 |
| Rework ($/pin) | $0.020 |
| QC ($/pin) | $0.004 |
| Power ($/watt) | $1.000 |
| Interconn | $0.020 |
| Program ($/part) | $0.250 |
| Caps. (each) | $0.005 |

(a)

**Integrated Circuits**

**Component Count:**

| Package | TTL | PLA | EPLD |
|---|---|---|---|
| DIP14 | 3 | | |
| DIP16 | 6 | | |
| DIP20 | | 3 | |
| DIP24 | | | 1 |

| ICs | Types |
|---|---|
| TTL | 4 |
| PLA | 2 |
| EPLD | 1 |

| Circuit Requirements: | $I_{CC}$ (max) | Interconnects |
|---|---|---|
| TTL circuit (total mA). | 160 | 52 |
| PLA circuit (total mA). | 360 | 14 |
| EPLD circuit (total mA). | 15 | 0 |

(b)

**Tables 10a and b. Window Circuit Cost Variables**

Table 11. Window Circuit Production Costs

| AVERAGE COMPONENT COST | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Year 1 | | | Year 2 | | | Year 3 | | |
| Package | TTL | PLA | EPLD | TTL | PLA | EPLD | TTL | PLA | EPLD |
| DIP14 | $0.22 | | | $0.19 | | | $0.17 | | |
| DIP16 | $0.44 | | | $0.37 | | | $0.26 | | |
| DIP20 | | $2.00 | | | $1.70 | | | $1.56 | |
| DIP24 | | | $6.00 | | | $4.20 | | | $2.90 |
| **PRODUCTION COSTS** | | | | | | | | | |
| | Year 1 | | | Year 2 | | | Year 3 | | |
| Item (costs per part) | TTL | PLA | EPLD | TTL | PLA | EPLD | TTL | PLA | EPLD |
| Components | $0.367 | $2.000 | $6.000 | $0.310 | $1.700 | $4.200 | $0.230 | $1.560 | $2.900 |
| Incoming Insp. | $0.153 | $0.200 | $0.240 | $0.153 | $0.200 | $0.240 | $0.153 | $0.200 | $0.240 |
| Inventory | | | | | | | | | |
| Maintenance | $0.029 | $0.038 | $0.045 | $0.029 | $0.038 | $0.045 | $0.029 | $0.038 | $0.045 |
| Storage | $0.032 | $0.042 | $0.050 | $0.032 | $0.042 | $0.050 | $0.032 | $0.042 | $0.050 |
| Processing | $0.231 | $0.347 | $0.520 | $0.231 | $0.347 | $0.520 | $0.231 | $0.347 | $0.520 |
| Printed Circuit Board | | | | | | | | | |
| Fabrication | $0.046 | $0.060 | $0.072 | $0.046 | $0.060 | $0.072 | $0.046 | $0.060 | $0.072 |
| Trace costs | $0.116 | $0.093 | $0.000 | $0.116 | $0.093 | $0.000 | $0.116 | $0.093 | $0.000 |
| Assembly | $0.096 | $0.125 | $0.150 | $0.096 | $0.125 | $0.150 | $0.096 | $0.125 | $0.150 |
| Board test | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 | $0.150 |
| Rework | $0.015 | $0.020 | $0.024 | $0.015 | $0.020 | $0.024 | $0.015 | $0.020 | $0.024 |
| QC | $0.061 | $0.080 | $0.096 | $0.061 | $0.080 | $0.096 | $0.061 | $0.080 | $0.096 |
| Power Supply | $0.089 | $0.600 | $0.075 | $0.089 | $0.600 | $0.075 | $0.089 | $0.600 | $0.075 |
| Total Cost/Part | $1.385 | $3.754 | $7.422 | $1.328 | $3.454 | $5.622 | $1.248 | $3.314 | $4.322 |
| | | | | | | | | | |
| Total Cost/System | $12.463 | $11.263 | $7.422 | $11.953 | $10.363 | $5.622 | $11.233 | $9.943 | $4.322 |
| Additional Costs/System | | | | | | | | | |
| Programming loss | $0.000 | $0.120 | $0.000 | $0.000 | $0.102 | $0.000 | $0.000 | $0.094 | $0.000 |
| Safety stock | $0.165 | $0.600 | $0.120 | $0.140 | $0.510 | $0.084 | $0.104 | $0.468 | $0.058 |
| Programming fee | $0.000 | $0.750 | $0.250 | $0.000 | $0.750 | $0.250 | $0.000 | $0.750 | $0.250 |
| De-coupling caps | $0.045 | $0.015 | $0.005 | $0.045 | $0.015 | $0.005 | $0.045 | $0.015 | $0.005 |
| True mfg. cost/system | $12.673 | $12.748 | $7.797 | $12.137 | $11.740 | $5.961 | $11.381 | $11.269 | $4.635 |

The production costs again show that the system cost for the first year is better with EPLDs. The two consecutive years show that the declining price of EPLDs make them an excellent candidate for systems that will ramp up production at that time.

## Window Circuit Conclusion

The TTL version of the circuit was implemented with MSI counters and decoders. As a result, the PAL implementation was bound by the number of count bits and had to be programmed into two PALs. In circuits like this, it is useful to rewire the decode for different counts depending on the application. The PAL implementation allows this by incorporating the decode and output latches into one IC.

The EPLD implementation tackles the MSI integration quite easily and also provides the capability to reprogram the decoder. Since the counter and output latches consist of fourteen registered outputs, the sixteen macrocells of the 5C060 easily accommodate the needed functions.

## SUMMARY

We have examined the hidden costs of production and how they differ for several logic alternatives. By examining these costs, we have shown that while an EPLD is presently a more expensive part, it's level of integration reduces system costs and improves reliability. The following items should be considered when evaluating logic alternatives:

- system cost is determined by more than component cost
- system cost and reliability is influenced by the type and amount of components used
- semiconductors have a life cycle that determines their present price at design, and at production time

In summary, when all system costs are considered, EPLDs can provide cost savings to the design and production of most board designs.

## REFERENCES


1. The Future Is Now: Extending CAE into test of custom VLSI.
   Robert S. Broughton, Tektronix.
   Michael G. Brashler, Tektronix.
   IEEE International Test Conference Proceedings, 1984
2. Reducing The Cost of Quality Through Test Data Managment.
   Paul N. Manikas, GenRad Inc.
   Stephen G. Eichenlaub, Harvard University.
   IEEE International Test Conference Proceedings, 1983
3. A Quantitative Analysis Of The Trade-offs Between Higher Capital Investment and Higher Yield In PCB Testing.
   Mark A. Myers, Teradyne Inc.
   IEEE International Test Conference Proceedings, 1984
4. An Analysis Of The Cost And Quality Impact Of LSI/VLSI Technology On PCB Test Strategies.
   Mark A. Myers, Teradyne Inc.
   IEEE International Test Conference Proceedings, 1983
5. IC Quality Control By The User.
   Roger Dunn, Xerox Corp.
   IEEE International Test Conference Proceedings, 1983
6. An Analysis Of The Economics of Self Test.
   P. Varma, University of Manchester.
   A. P. Ambler, University of Manchester.
   K. Baker, GEC Research Labs.
   IEEE International Test Conference Proceedings, 1984
7. In Circuit Testability Factors: Shoot With A Rifle.
   Douglas W. Raymond, Zehntel Production Services.
   IEEE International Test Conference Proceedings, 1984
8. Seven Steps To Zero Defects.
   D. W. Rudd, AT&T Technologies.
   Circuits Manufacturing, June 1986
9. Rework Forum
   Donald Ford, Senior Editor.
   Circuits Manufacturing, September 1986
10. Manufacturing Defect Analyzers: Annual Roundup.
    Evaluation Engineering magazine, August 1986
11. Assembly: Automation Makes It Better.
    Roland W. Roy and Gordon Weeks, Andover Controls.
    Circuits Manufacturing, February 1986
12. Shrinking Lines Squeeze Processes.
    Jerry Murray, West Coast Editor.
    Circuits Manufacturing, September 1986
13. Ribbon Cable for Reliable Interconnections.
    Bennett W. Brachman, Xport Trading Inc.
    Electronic Packaging and Production magazine, July 1986

14. TQC and JIT: Partners In Production.
Rick Walleigh, Hewlett Packard.
Circuits Manufacturing, February 1986

15. Automated Handling/Sorting: Multisite Development Moves to Back Burner.
Evaluation Engineering magazine, May 1986

16. Software Charts The Course of Component Testing.
Ronald Pound, Editor.
Electronic Packaging and Production magazine, June 1986

17. Complexity, PLDs Drive The Market.
Evaluation Engineering magazine, July 1986

18. Intel User Defined Logic Handbook.
Intel Corp. 1986

19. VLSI Semicustom Design Guide.
CMP Publications, Summer 1986

20. AMD Programmable Array Logic Handbook.
Advanced Micro Devices, 1984

21. Handbook Of Industrial Engineering.
Gavriel Salvendy, Editor, Purdue University
John Wiley & Sons Publications

22. Components Quality/Reliability Handbook.
Intel Corporation.

23. The Cost Edge.
DM DATA Corp.
Scottsdale, AZ

24. Semiconductor Purchasing Strategies Integrated Circuits Engineering Corp.
Scottsdale, AZ

25. Status 1986 Integrated Circuits Engineering Corp.
Scottsdale, AZ

26. EDN Semicustom Design Series
EDN Magazine, 1985

27. EDN Design Ideas
EDN Magazine, 1985